METHOD AND SYSTEM FOR MULTIDIMENTIONAL DATABASE MANAGEMENT

Field of the Invention

The current invention is generally related to multidimensional database management, and more particularly related to processing multidimensional data without ~~layer~~ layer or hierarchy structure information.

BACKGROUND OF THE INVENTION

A multidimensional database model contains data corresponding to a point having values in multidimensional space that is defined by a plurality of dimensions. The multidimensional database model is generally effective in analyzing the data in a multifaceted manner. Based upon a predetermined rule, it is a basic function to correspond a value of data that corresponds to one or more of points in the multidimensional space to other values corresponding to other points. For example, the above basic function is disclosed in "OLAP Practical Data Warehouse," Toyoshima and Kimura, pp76-79, (1997); "OLAP Solutions Building Multidimensional Information Systems," Thomsen, pp. 89-104, (1997). The above described basic function is useful in constructing a database with minimally necessary data input as well as in retrieving data that has been already calculated.

In a multidimensional database model, the multidimensional space as defined by a plurality of dimensions is called "cube." The dimensions in the cube is generally expressed by a set of members that have a layer structure. A point in the multidimensional space as defined by an arbitrary member at each dimension is named "cell," and a value of the data corresponding to the cell is "a cell value." For example, using multidimensional data that represents sales units and sales amounts of products sold by a company A in

Japan, the cube includes a time dimension, a retail store dimension, a merchandise dimension and a ~~unit dimension~~unit or measure dimension.

To illustrate the above related example, FIGURE 20 is a diagram for describing conventional layer structure information storing layer information on members for each dimension. A ~~layer~~ layer or hierarchy 4301 for the time dimension has "1999" as the highest member or the highest member in the ~~layer~~ layer or hierarchy structure, and the highest member 1999 in turn has two child members, 1999Q1 and 1999Q2. Similarly, the child member 1999Q1 has grand child members, 199901, 199902 and 199903. By the same token, the other child member 1999Q2 has grand child members, 199904 and 199905. The grand children members, 199901, 199902, 199903, 199904 and 199905 do not have any great grand children and are defined as the lowest members in the above example. The level is accordingly assigned to the members based upon a position in the layer. The lowest members are situated at Level 0 while the direct parent of the lowest members is located at Level 1. Similarly, the direct parent of Level 1 members are located at Level 2. The highest members are located at the level that is equal to the number of layers minus one.

Still referring to FIGURE 20, a layer 4302 for a retail store dimension has all the sales territories or nation wide territories as the highest member. The highest member has two child members including Eastern Japan and Western Japan. Eastern Japan further owns two child members including Chiba store and Saitama store. Western Japan further owns two child members including Osaka store and Hiroshima store. Similarly, a layer 4303 for a merchandise dimension has the all merchandises as the highest member. The highest member has two child members including home appliance and audiovisual equipment (AV). Home Appliance further owns two child members including washers and refrigerators. AV further owns two child members including television sets and video equipment. Lastly, a layer 4304 in a unit dimension has two members including a number of sales and an amount of sales but, the members in the unit dimension have no layer structure.

Now referring to FIGURE 21, a diagram illustrates an example of conventional layer structure definition data. The conventional layer structure definition data 3201 through 3204 respectively define the time dimension, the retail store dimension, the merchandise dimension and the unit dimension, and each of the conventional layer structure definition data 3201 through 3204 contains a plurality of records that are ~~separated by lines~~separated by new line characters. Each of the records has members that are separated by commas, and the separation format is called comma separated value format (CSV). One record potentially includes all the members from the highest member to the lowest member. The conventional layer structure definition data 3201 through 3203 each is organized to list records according to the level. That is, in the above example, a record at Level 2 is followed by a record at Level 1 and then by a record at Level 0. For the layer structure definition data 3204, a record has only members at Level 0.

Now referring to FIGURE 22, a diagram illustrates an example of conventional data. The data 3301 is stored in the cube and in the above described CSV format. Each of the record includes members and corresponding cell values. In the example, each record thus contains five fields respectively for information on month/year, a retail store, a merchandise name, a number of sales and an amount of sales. The first three fields are respectively from the time dimension, the retail store dimension and the merchandise dimension. On the other hand, the last two fields are cell values representing the number of sales and the amount of sales from the unit dimension.

FIGURE 23 is a diagram illustrating an exemplary display of multidimensional data. The exemplary display is a screen multidimensional data analysis on a terminal device. The horizontal axis includes members on the time dimension while the vertical axis includes members on the retail store dimension. All members are displayed from the above dimensions. On the other hand, as shown in the upper right corner of the display, only one member such as television is displayed in the merchandise dimension and only one member such as sales amount is displayed in the unit dimension. The above

exemplary screen display shows the sales amount for televisions at each retail store for each quarter. For example, the data representing the member 1999Q1 for the time dimension is a total value of the data 199901, 199902 and 199903. Similarly, the data representing the member, Eastern Japan for the time dimension is a total value of the data Chiba store and Saitama store. According to the previously noted dimensional layer structure, the data is a total value. One of the basic functions of the multidimensional database management system is to search the total value. To obtain more detailed data, other basic functions include a drill down function for changing a currently displayed member to its child member or a drill up function for conversely changing the currently displayed member to its parent member.

In order to respond to a search request for a total value at a high speed, other multidimensional database functions store previously calculated total values on an as-necessary basis. For example, Japanese Patent Publication Hei 9-265479 discloses techniques to search and ~~summarize~~ aggragate multidimensional data at a high speed. Another example, Japanese Patent Publication Hei 11-~~22461~~ 224261 discloses storage and search techniques for data having discontinuous members in a multidimensional database. There are relatively frequent needs to store in a multidimensional database some member data that is not included in the current layer and then to analyze the data. Using the example of sales data, the above described data includes a new store to be opened and a new product to be sold, and the new data is to be included in the multidimensional data. To analyze sales data for the new products such as books and music CDs that have new titles almost on a daily basis, the multidimensional data layer needs to be updated at a high frequency.

Now referring to FIGURE 24, a diagram illustrates conventional exemplary data that includes members that are not registered in the layer structure information. For example, a first field of a first record is a member 3311 that has a value of "199906." The member 3311 is unregistered in the time dimensional layer 4301 as shown in FIGURE 20. Similarly, a second field of a second record is a member 3312 that has a value of

"Yokohama Store." The member 3312 is unregistered in the retail store dimensional layer 4302 as shown in FIGURE 20. Lastly, a third field of a third record is a member 3313 that has a value of "~~refrigerator~~PCs." The member 3313 is unregistered in the merchandise dimensional layer 4303 as shown in FIGURE 20. In general, there are two ways to process the above described unregistered member data. The first way is to regard the unregistered member data to be invalid, and its corresponding record that contains the unregistered member data is also regarded as being invalid. The second way is to regard the unregistered member data to be new.

The above two options are further related to the database operations. The first processing option terminates the data handling operation upon detecting a record containing any unregistered member data. Alternatively, the data handling operation skips the record to a next one according to the first processing option. The data handling operation terminates by issuing an alarm signal, and the unregistered member remains to be unregistered at the layer information. Of course, the corresponding record is not included in the database and remains excluded from analysis. On the other hand, the above second processing option registers the currently unregistered member. In registering the new member, there are two ways to find a new position in the layer. One position is created at a new location as a new member without having any relation to the existing members.

Now referring to FIGURE 25, a diagram illustrates layer structure information for conventional data that includes members that are not previously registered in the layer structure information. The layer 4305 indicates that an unregistered member, "Yokohama store" is now registered as an independent member 4306 in the layer information structure 4305. The above described processing allows the incorporation of the unregistered member 4306 into the layer of the multidimensional database. Despite the incorporation of the previously unregistered member 4306, during an analysis stage, since the newly incorporated member 4306 is isolated from other members, the isolated member 4306 is not reflected in the summary. Furthermore, since the isolated member 4306 cannot be

FIGURE 26 is a diagram illustrating a portion of retail store dimensional layer rule definition data that is used in the fifth preferred embodiment according to the current invention.

FIGURE 27 is a diagram illustrates a portion of data that is to be stored in the multidimensional database in the fifth preferred embodiment according to the current invention.

FIGURE 28 is a flow chart illustrating steps involved in a fifth preferred process of the layer structure information updating process according to the current invention.

FIGURE 29 illustrates that the change from the retail store dimension layer 4302 as shown in FIGURE 20 to the final retail store dimension layer 4330.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT(S)

Referring now to the drawings, wherein like reference numerals designate corresponding structures throughout the views, and referring in particular to FIGURE 1, a diagram illustrates a preferred embodiment of the multidimensional database processing system according to the current invention. In the specification of the current application, the term, "layer" is interchangeably used with the term, "hierarchy" or "hierarchical" to have the substantially identical meaning. Similarly, the terms, "character row conversion" and "character string replacement" are interchangeably used in the current application. The multidimensional database processing system includes a computer system 1 that includes a central processing unit (CPU) 2, a ~~main memory~~ main or primary memory unit 3, a secondary or external memory unit 4 such as magnetic disks and a plurality of terminals 6 that are connected by a network 5. A multidimensional database management unit 10 includes a system control unit 11, a multidimensional data management unit 12, a dimensional ~~layer~~ layer or hierarchy information management unit 13, a ~~layer~~ layer or

hierarchy rule management unit 14, a ~~layer~~ layer or hierarchy structure information management unit 15 and a ~~layer~~ layer or hierarchy structure information update unit 16. The multidimensional database management unit 10 exists in the main memory unit 3 and includes the system control unit 11, the multidimensional data management unit 12 and the dimensional layer information management unit 13. The system control unit 11 controls the system in general. In particular, the system control 11 receives a layer rule definition request 21, a multidimensional data storage request 22 and a multidimensional data analysis request 23 and sends the requests 21, 22 and 23 to the multidimensional data management unit 12 and the dimensional layer information management unit 13. The system control 11 subsequently sends the processed results back to the requested terminals 6. The multidimensional data management unit 12 generally manages a multidimensional database 44 in an external or secondary memory unit 4. The multidimensional data management unit 12 has a first function to read storage data 33 and to store it in the multidimensional database 44, a second function to search in the multidimensional database 44 and to return the multidimensional data as well as a third function to return the data that is specified for an analysis by the search range in the multidimensional database 44.

Still referring to FIGURE 1, the dimensional layer information management unit 13 further includes the layer rule management unit 14, the layer structure information management unit 15 and the layer structure information update unit 16. The dimensional layer information management unit 13 manages dimensional layer information that exists in the externally located secondary memory unit 4. The layer rule management unit 14 manages layer rule 42 that is used in generating layer information for a member. The layer rule management unit 14 has functions to read layer structure rule definition data 31, to register at the layer rule 42 and to obtain the layer rule 42. The layer structure information management unit 15 manages layer structure information 43 for registering members in a dimension. The layer structure information management unit 15 has functions to read the layer structure definition data 32, to search layer structure information 43 and to obtain the layer rule 42 if layer information on the specified member is not registered at the layer

generation method for each member at LEVEL 0, LEVEL 1 and LEVEL 2. In particular, the fourth line indicates that the member name is used without altering at the lowest member or LEVEL 0. The fifth line through the tenth line indicate the generation method for LEVEL 0. The sixth line through the ninth line specify replacement or conversion rules based on formal expressions. A new member name at LEVEL 2 is obtained by applying a character row conversion as expressed by the formal expressions to an input member name. For example, the formal expression on the seventh line is used to describe the above character row conversion process. If the fifth and sixth characters in the character row are "04", "05" or "06," a character row, "Q2" replaces them after the first or fourth character of the input member to obtain the character row conversion output. Assuming that an input member name is "199906," the LEVEL 1 member name becomes "1999Q2." The eleventh and thirteenth lines indicate that a row of the first through fourth characters is used without modification as a LEVEL 2 member name. For example, if an input member name is "199906," "1999" becomes the LEVEL 2 member name.

Now referring to FIGURE 9, a diagram illustrates exemplary layer structure definition data for unregistered members as used in a preferred embodiment according to the current invention. The exemplary layer structure definition data is generated for storing data 3310 including unregistered data of FIGURE 24. When the layer information is generated for the unregistered member 3311 of FIGURE 24 based upon the layer rule definition data 3110 as shown in FIGURE 8, the layer structure definition data 3211 is generated as layer information. Layer rule definition data for generating the layer structure definition data 3212 or ~~3212~~ 3213 will be described later.

FIGURE 10 is a diagram illustrating an example of the layer structure information 43 after being modified by a preferred embodiment according to the current invention. Layers 4311 through 4313 show the layer structure information 43 after the layer information of the unregistered members 3311 through 3313 of FIGURE 24 is registered using the layer structure definition data 3211 through 3213 of FIGURE 9. For example, the layer 4301 as shown in FIGURE 20 is registered at the layer structure

database management unit 10 is further connected to the dimensional layer information ~~unit~~ 41 and the multidimensional database 44 as shown in the first preferred embodiment with respect to FIGURE 1. The dimensional layer information ~~unit~~ 41 further includes the layer rule 42 as well as the layer structure information 43.

Now referring to FIGURE 15, a diagram illustrates a portion of merchandise dimensional layer rule definition data 3130 that is used in the third preferred embodiment according to the current invention. The exemplary layer rule 42 includes the above layer rule definition data 3130. The first two lines in the layer rule definition data 3130 are comments. The third line indicates that the layer information is to be generated from a data table named "~~RDB Product Master List~~Product Master Table." The fourth and fifth lines respectively indicate the correspondence of the LEVEL 0 members and LEVEL 1 members in the above specified list. The sixth line indicates that the LEVEL 2 member is "All Products" or "All Merchandise."

Now referring to FIGURE 16, a diagram illustrates an exemplary RDB product master table for the RDB that is used in the third preferred embodiment according to the current invention. The RDB product master table stores member names on the merchandise dimension in the RDB 26. In combination with the information on the fourth and the following lines in the layer rule definition data 3130 of FIGURE 15, the layer information is generated for the merchandise dimension. For example, given an unregistered member, "PCs," the RDB 26 is searched to obtain a record 261 whose data value for the small classification is "PCs." The ~~fourth~~ fifth and the ~~fifth~~ sixth lines in the layer rule definition data 3130 of FIGURE 15 specify the LEVEL 1 and LEVEL 2 members that correspond to "computers" and "all merchandises."

Now referring to FIGURE 17, a flow chart illustrates steps involved in a third preferred process of the layer structure information updating process 90B according to the current invention. The layer rule 42 and the above specified member from the layer structure information management unit 15 are inputted to the layer structure information

updating process 90B. In a step 9021, the layer structure information update unit 16 receives an instruction for layer information generation for the above specified member from the layer structure information management unit 15 and examines the layer rule 42. If ~~the layer information is not to be generated from the RDB 26~~the hierarchy rule 42 is not defined that the hierarchical information is to be generated from RDB 26, the third preferred process terminates. On the other hand, if the layer information is to be generated from the RDB 26, the third preferred process branches to a step 9022. In the step 9022, the RDB management system 25 is connected based upon a predetermined procedure. In a step 9023, an inquiry is made to the RDB 26 based upon the above specified ~~members~~ members and the parameters that are specified in the layer rule 42. In the step 9024, the layer structure definition data is generated as the layer ~~information~~ information of the specified member from the inquiry results according to the layer rule 42. In the step 9024, by using the generated layer structure definition data, the layer information of the specified member is also stored in the layer structure information 43, and the above generated layer information is returned to from the layer structure information management unit 15. The third preferred process then terminates.

As described above, in the third preferred process, the multidimensional database improves the management efficiency. For example, the layer 4303 as shown in FIGURE 20 is registered at the layer structure information 43 of FIGURE 1 and the retail store dimension layer rule definition data 3130 as illustrated in FIGURE 15 is registered in the layer rule 42 of FIGURE 1. When the member 3313 of the third record in the layer rule definition data 3130 is to be stored in the multidimensional database 44 of FIGURE 1, it is detected that the member 3313 or "PCs" is unregistered. Upon the detection, the layer rule definition data 3130 is used to generate the layer structure definition data 3213 as shown in FIGURE 9 for the above unregistered member, and the newly generated layer structure definition data 3213 is stored at the layer structure information 43. As a result, the merchandise dimensional layer 4302 as shown in FIGURE 20 changes to the retail store dimensional layer 4313 as shown in FIGURE 10. As described above, the preferred embodiment according to the current invention generates the layer information for a

WHAT IS CLAIMED IS:

1. A method of maintaining a multidimensional database having layer structure information, the layer structure information representing layer structure of members in the multidimensional database, comprising the steps of:

inputting a member;

determining whether or not corresponding layer information has been registered for the member in the layer structure information;

in the absence of the corresponding information, generating the corresponding layer information according to a predetermined layer rule; and

registering the corresponding layer information in the layer structure information.

2. The method of maintaining a multidimensional database according to claim 1 wherein the predetermined layer rule includes rules for character-row converting a name of the member and for generating the layer information.

3. The method of maintaining a multidimensional database according to claim 2 wherein the rules are expressed in ~~formal expressions~~regular expression.

4. The method of maintaining a multidimensional database according to claim 1 wherein the predetermined layer rule includes ~~a name~~an identifier of a file and descriptions of a predetermined format, the file containing ~~predetermined rules~~a source for generating the corresponding layer information.

5. The method of maintaining a multidimensional database according to claim 4 wherein said generating step further comprising a step of accessing the file to obtain ~~the predetermined rules~~hierarchical information.

6. The method of maintaining a multidimensional database according to claim 1 wherein the predetermined layer rule includes ~~a name~~an identifier of a database and

descriptions of a predetermined format, the database containing ~~predetermined rules~~a source for generating the corresponding layer information.

7. The method of maintaining a multidimensional database according to claim 6 wherein said generating step further comprising a step of accessing the database to obtain the ~~predetermined rules~~source.

8. The method of maintaining a multidimensional database according to claim 1 further comprising additional steps of:

determining whether or not the layer structure information exists before said inputting the member; and

in the absence of the layer structure information, generating the layer structure information that represents the layer structure of the members in the multidimensional database.

9. The method of maintaining a multidimensional database according to claim 1 wherein the predetermined layer rule further comprises a main layer rule and a plurality of sub-layer rules and further comprising additional steps of:

generating the corresponding layer information according to a sequential application of the main layer rule and the plurality of the sub-layer rules; and

determining whether or not the layer information is successfully generated.

10. The method of maintaining a multidimensional database according to claim 1 further comprising additional steps of:

inquiring about the layer information for the member in the layer structure information from a calling unit prior to said inputting step; and

returning the layer information to the calling unit subsequent to said registering step.

11. A system for maintaining a multidimensional database, comprising:

an input unit for inputting a member;

a dimensional layer information unit for storing dimensional layer information, the dimensional layer information including layer structure information to represent layer structure of members in the multidimensional database and a layer rule for defining rules to generate the layer structure information; and

a dimensional layer information management unit connected to said input unit and said dimensional layer information unit for managing the dimensional layer information, said dimensional layer information management unit determining whether or not corresponding information has been registered for the member in the layer structure information, said dimensional layer information management unit generating the corresponding layer information according to the layer rule in the absence of the corresponding information and registering the corresponding layer information in the layer structure information.

12. The system for maintaining a multidimensional database according to claim 11 wherein the layer rule includes rules for character-row converting a name of the member and for generating the layer information.

13. The system for maintaining a multidimensional database according to claim 12 wherein the rules are expressed in ~~formal expressions~~regular expression.

14. The system for maintaining a multidimensional database according to claim 11 wherein the predetermined layer rule includes ~~a name~~an identifier of a file and descriptions of a predetermined format, the file containing ~~predetermined rules~~a source for generating the corresponding layer information.

15. The system for maintaining a multidimensional database according to claim 14 wherein said dimensional layer information management unit accesses the file to obtain ~~the predetermined rules~~hierarchical information.

16. The system for maintaining a multidimensional database according to claim 11 wherein the predetermined layer rule includes ~~a name~~an identifier of a database and descriptions of a predetermined format, the file containing ~~predetermined rules~~a source for generating the corresponding layer information.

17. The system for maintaining a multidimensional database according to claim 16 wherein said dimensional layer information management unit accesses the database to obtain the ~~predetermined rules~~source.

18. The system for maintaining a multidimensional database according to claim 11 wherein said dimensional layer information management unit determines whether or not the layer structure information exists before inputting the member, in the absence of the layer structure information, said dimensional layer information management unit generating the layer structure information that represents the layer structure of the members in the multidimensional database.

19. The system for maintaining a multidimensional database according to claim 11 wherein the predetermined layer rule further comprises a main layer rule and a plurality of sub-layer rules, said dimensional layer information management unit generating the corresponding layer information according to a sequential application of the main layer rule and the plurality of the sub-layer rules, said dimensional layer information management unit determining whether or not the layer structure information is successfully generated.

20. The system for maintaining a multidimensional database according to claim 11 wherein said input unit inquiring the layer information for the member to said dimensional layer information management unit, said dimensional layer information management unit returning the layer information to said input unit subsequent to registering the layer information.

21. A recording medium for storing computer executable instructions for maintaining a multidimensional database having layer structure information, the layer structure information representing layer structure of members in the multidimensional database, the computer executable instructions performing the steps of:

inputting a member;

determining whether or not corresponding layer information has been registered for the member in the layer structure information;

in the absence of the corresponding information, generating the corresponding layer information according to a predetermined layer rule; and

registering the corresponding layer information in the layer structure information.

22. The recording medium for storing computer executable instructions according to claim 21 wherein the predetermined layer rule includes rules for character-row converting a name of the member and for generating the layer information.

23. The recording medium for storing computer executable instructions according to claim 22 wherein the rules are expressed in ~~formal expressions~~regular expression.

24. The recording medium for storing computer executable instructions according to claim 21 wherein the predetermined layer rule includes ~~a name~~an identifier of a file and descriptions of a predetermined format, the file containing ~~predetermined rules~~a source for generating the corresponding layer information.

25. The recording medium for storing computer executable instructions according to claim 24 wherein said generating step further comprising a step of accessing the file to obtain ~~the predetermined rules~~hierarchical information.

26. The recording medium for storing computer executable instructions according to claim 21 wherein the predetermined layer rule includes ~~a name~~an identifier of

a database and descriptions of a predetermined format, the database containing ~~predetermined rules~~a source for generating the corresponding layer information.

27. The recording medium for storing computer executable instructions according to claim 26 wherein said generating step further comprising a step of accessing the database to obtain the ~~predetermined rules~~source.

28. The recording medium for storing computer executable instructions according to claim 21 further comprising additional steps of:

determining whether or not the layer structure information exists before said inputting the member; and

in the absence of the layer structure information, generating the layer structure information that represents the layer structure of the members in the multidimensional database.

29. The recording medium for storing computer executable instructions according to claim 21 wherein the predetermined layer rule further comprises a main layer rule and a plurality of sub-layer rules and further comprising additional steps of:

generating the corresponding layer information according to a sequential application of the main layer rule and the plurality of the sub-layer rules; and

determining whether or not the layer information is successfully generated.

30. The recording medium for storing computer executable instructions according to claim 21 further comprising additional steps of:

inquiring about the layer information for the member in the layer structure information from a calling unit prior to said inputting step; and

returning the layer information to the calling unit subsequent to said registering step.